

Patent claims

SUB A1

1. A method for the computer-aided elimination of at least one inconsistency in a database collection containing a database and at least one copy database of the database, which inconsistency arises on account of the database and/or the copy database being changed,

5 a) in which at least some of the operations which create an inconsistency are allocated to defined conflict types,

10 b) in which each conflict type is allocated a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created by at least one operation of the respective conflict type,

15 c) in which the inconsistency is eliminated using the decision set.

2. The method as claimed in claim 1, in which a plurality of inconsistencies are eliminated.

20 3. The method as claimed in claim 1 or 2, in which each conflict type is allocated a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created by a plurality of operations of the respective conflict type.

25 4. The method as claimed in one of claims 1 to 3, in which the database collection contains a plurality of copy databases of the database.

5. The method as claimed in one of claims 1 to 4, 30 in which all the inconsistencies and their dependencies on one another are ascertained before the inconsistencies are eliminated.

6. The method as claimed in one of claims 1 to 5, in which a conflict, an anomaly or a pseudo-anomaly is 35 ascertained when an inconsistency is ascertained.

7. The method as claimed in one of claims 1 to 6, in which, during elimination of the inconsistencies, the decision set for at least one conflict type is modified depending on the dependencies of the

~~inconsistencies.~~

8. The method as claimed in one of claims 1 to 7, in which, after a prescribable number of eliminated inconsistencies, the database collection is examined 5 for further inconsistencies and their dependencies, anomalies and pseudo-anomalies.

9. The method as claimed in one of claims 1 to 8, in which the database collection contains an object-oriented database.

10. 10. The method as claimed in one of claims 1 to 9, used within the context of object-oriented software development.

11. The method as claimed in one of claims 1 to 9, used within the context of creating a structured 15 electronic document.

12. An arrangement for eliminating at least one inconsistency in a database collection containing a database and at least one copy database of the database, which inconsistency arises on account of the 20 database and/or the copy database being changed, having at least one processor which is set up such that the following steps can be carried out:

- 25 a) at least some of the operations which create an inconsistency are allocated to defined conflict types,
- b) each conflict type is allocated a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created by at least one operation of the respective 30 conflict type,
- c) the inconsistency is eliminated using the decision set.

13. The arrangement as claimed in claim 12, in which the processor is set up such that a plurality 35 of inconsistencies are eliminated.

14. The arrangement as claimed in claim 12 or 13, in which the processor is set up such that each conflict type is allocated a decision set which is used to indicate possible decisions which can be used to

- 52 -

eliminate an inconsistency created by a plurality of operations of the respective conflict type.

15. The arrangement as claimed in one of claims 12 to 14,

5 in which the processor is set up such that the database collection contains a plurality of copy databases of the database.

16. The arrangement as claimed in one of claims 12 or 15,

10 in which the processor is set up such that all the inconsistencies and their dependencies on one another can be ascertained before the inconsistencies are eliminated.

17. The arrangement as claimed in one of claims 12 to 16,

15 in which the processor is set up such that a conflict, an anomaly or a pseudo-anomaly can be ascertained when an inconsistency is ascertained.

18. The arrangement as claimed in one of claims 12 to 17,

20 in which the processor is set up such that, during elimination of the inconsistencies, the decision set for at least one conflict type can be modified depending on the dependencies of the inconsistencies.

25 19. The arrangement as claimed in one of claims 12 to 18,

30 in which the processor is set up such that, after a prescribable number of eliminated inconsistencies, the database collection is examined for further inconsistencies and their dependencies, anomalies and pseudo-anomalies.

35 20. The arrangement as claimed in one of claims 12 to 19,

in which the processor is set up such that the database collection contains an object-oriented database.

21. The arrangement as claimed in one of claims 12 to 20,

used within the context of object-oriented software development.

- 53 -

22. The arrangement as claimed in one of claims 12 to 21,
used within the context of creating a structured electronic document.

5 23. A set of a plurality of arrangements for eliminating at least one inconsistency in a database collection containing a database and at least one copy database of the database, which inconsistency arises on account of the database and/or the copy database being 10 changed,
in which each arrangement has at least one processor which is set up such that the following steps can be carried out:

15 a) at least some of the operations which create an inconsistency are allocated to defined conflict types,

b) each conflict type is allocated a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created 20 by at least one operation of the respective conflict type,

c) the inconsistency is eliminated using the decision set, and
in which the arrangements can be coupled to one 25 another.

ADD A27